Europäisches Patentamt
European Patent Office
Office européen des brevets

Bescheinigung Certificate Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. Patent application No. Demande de brevet n°

01650060.5

Der Präsident des Europäischen Patentamts:
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

I.L.C. HATTEN-HECKMAN

THIS PAGE BLANK (USPTO)

Anmeldung Nr:
Application no.: 01650060.5
Demande no:

Anmeldetag:
Date of filing: 23.05.01
Date de dépôt:

Anmelder/Applicant(s)/Demandeur(s):

Markport Limited
Custom House Plaza 5
Harbourmaster Place
Dublin 1
IRLANDE

Bezeichnung der Erfindung/Title of the invention/Titre de l'invention:
(Falls die Bezeichnung der Erfindung nicht angegeben ist, siehe Beschreibung.
If no title is shown please refer to the description.
Si aucun titre n'est indiqué se referer à la description.)

An applications framework for network communication

In Anspruch genommene Priorität(en) / Priority(ies) claimed /Priorité(s) revendiquée(s)
Staat/Tag/Aktenzeichen/State/Date/File no./Pays/Date/Numéro de dépôt:

Internationale Patentklassifikation/International Patent Classification/
Classification internationale des brevets:

H04L29/00

Am Anmeldetag benannte Vertragstaaten/Contracting states designated at date of filing/Etats contractants désignées lors du dépôt:

AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE TR

"An applications framework for network communication"

Introduction

The invention relates to providing access to applications and content from networks, particularly but not exclusively mobile networks.

At present, it is often the case that a new access mechanism is needed for each new user device technology. This results in a requirement for skilled developers and in use of toolkits and IDEs which increase commitment to a single technology. Conversion systems have been developed to address these problems, for example Web-based automatic translation of HTML to WAP or CHTML. However this approach often provides only short term benefits and also suffers from the problem of allowing a low degree of interactivity. Also, the linkages through the translation functions are relatively easy to break. In another approach, there is XML-centric automatic translation in which XML is translated into formats such as WAP or HTML. However, this approach does not provide support for different degrees of interactivity, and there is little user context support.

The invention addresses these problems.

Statements of Invention

According to the invention, there is provided an applications framework comprising:

an adapter comprising means for interfacing with a user mobile device;

an access broker comprising means for routing requests and responses according to device technology and content attributes; and

an application platform comprising means for allowing access to content and application servers via the access broker.

In one embodiment, the adapter comprises means for managing the context for a device making a request, and for converting an input into a Web request using input data, device context, and application context.

In another embodiment, the adapter comprises means for adding user and location context to requests.

In another embodiment, the adapter comprises means for translating responses into device-specific format using response date, device context, and application context.

In one embodiment, the adapter comprises means for updating and storing context between device interactions.

In another embodiment, the framework comprises a plurality of adapters, each associated with a type of mobile device.

In one embodiment, the access broker comprises means for controlling access to Web applications according to user subscription, in which responses are split up and routed according to adapter capabilities, content attributes, and user-specified rules.

In another embodiment, the access broker comprises means for managing a register of adapter capabilities and of currently accessible adapters for each user.

In a further embodiment, the access broker comprises means for translating service values placed by applications, and for routing the translated data to external systems.

In one embodiment, the application platform comprises means for an API to allow alternative interfaces for interactive applications.

In another embodiment, the adapter, the access broker, and the application platform comprises means for communicating with each other using an XML-compliant markup language.

In one embodiment, content is defined in elements, in which a root element is an abstraction of a mobile device screen.

In another embodiment, wherein sound streams and images are defined as elements.

Detailed Description of the Invention

The invention will be more clearly understood from the following description of some embodiments thereof, given by way of example only with reference to the accompanying drawings in which:-

Fig. 1 is a diagrammatic representation of an applications framework;

Fig. 2 is a diagram illustrating the range of user devices which may communicate via the framework;

Fig. 3 is a diagram illustrating an adapter in more detail;

Fig. 4 is a diagram illustrating a broker in more detail; and

Fig. 5 is a diagram illustrating micro device applications.

Referring to Fig. 1 an applications framework 1 comprises a central access broker 2, a technology adapter 3, and an applications platform 4. There may be more than one technology adapter and applications platform.

The technology adapter 3 performs the following main functions.:-

manage context for an individual device,

augment user input data with device context and application context,

convert an input into a request using input data, device context, and application context,

add user and location context to requests,

translate responses into device-specific format using response data, device context, and application context, and

updates and store context between device interactions.

The access broker 2 performs the following functions:-

control of access to framework-based applications,

handles user subscriptions and billing,

split-up and route responses according to technology adapter capabilities, content attributes, and user-specified rules, and

management of register of adapter capabilities and currently accessible adapters for each user.

Regarding the billing function above, the access broker 2 allows framework applications to associate a value, either monetary or not with messages they deliver to a micro device. This value reflects the charge that should be levied on the

recipient for viewing that message. The access broker can translate this information into a format that can be used by external billing and electronic commerce systems.

The applications platform 4 provides the following functions:-

HTTP request/response mechanism,

Framework-specific XML content handling,

use of JSP to generate content dynamically, and

API providing alternative interface for interactive applications.

In more detail, Fig. 2 illustrates capabilities and connectivities of devices which can use the framework 1. The devices are referred to as "micro" devices because they are small, more "personal", and having lower levels of processing power than traditional counterparts in the fixed-line Internet domain. It will be appreciated from Fig. 2 that there is a wide range of both capabilities and access protocols. It can be seen from Fig. 2 that micro device capabilities can be characterised by:

Display Characteristics. Displays range from simple text only LCD displays in low-end handsets through to high resolution, high colour TFT displays in laptop computers. In the future, there may well be micro devices that use speech recognition and speech synthesis access and deliver Wireless Internet content – this may even become many users preferred user interface.

Processing Ability. In most cases, phone-based micro devices have only a limited ability to run scripts and decode compressed and encrypted data. However, PDAs and laptop computers have processors many times more powerful and are capable of decoding streaming multimedia feeds and complex security protocols.

Storage Ability. Typically, phone based micro devices are built with limited data storage capabilities. PDAs and smart phones may have several megabytes and laptop computers may have several gigabytes.

Connectivity. Presently, micro devices are using SMS, USSD and CSD technologies to connect to the Wireless Internet. Data transfer rates are extremely slow and the device only sets up a data connection when initiated by the user. Shortly we will see devices using 2.5G technologies such as GPRS and EDGE working at much higher data rates. Many of these devices will offer an 'always on' capability, enabling them to receive Wireless Internet content without the user having to actively establish a data session. In the future, we will see new 3G devices that offer even greater data transfer rates. These devices may well be true IP devices offering an 'always on' connection directly to the Internet.

As the Wireless Internet grows and attracts an ever more diverse set of users with differing requirements it is likely that we will see the range of micro devices expanding in response. Without a structured approach, providing compatible Wireless Internet services is going to become increasingly more difficult..

Referring again to Fig. 1, the layers communicate with each other using the Micro Device Mark-up Language (MDML). MDML is a proprietary XML compliant format for defining multimedia content and associated attributes.

The adaption layer 3 is made up of a number of Technology Adaptors, each responsible for allowing services and content to be delivered to a specific class of Micro Device, examples include:

- SMPP, OIS and EMI protocols for SMS via Logica, Sema and CMG SMSCs respectively
- SMPP for USSD via Logica's USSD Server

- WML for WAP devices
- HTML
- cHTML for i-Mode phones
- SMTP for e-mail delivery

Each of these device classes has different capabilities and different methods of connecting to the Wireless Internet. The primary function of each of the technology adapters is to 'shelter' application and content developers from these differences, allowing them to work with a standard, micro device, interface. In order to do this, technology adapters for less capable micro devices (such as SMS-only phones) build this additional functionality into the network. As new classes of micro device come to market, new technology adapters will be created. This will allow new devices to access existing services without the need to redevelop and test existing content and application code.

The access layer is controlled by the access broker 2. The access broker 2 performs a number of tasks, including:

- Maintenance of a subscriber register, the services to which they have subscribed and their preferences
- Policing of user access to applications and content, ensuring users are only able to access services to which they have subscribed
- Where a subscriber has access to a range of micro devices, routing content to the most appropriate
- Interface to billing and advertisement engines

The application layer is where Wireless Internet applications and content resides. Fixed content is formatted as MDML Web pages that are provided to the access broker by a standard Web Server platform. Dynamic content is provided using industry standard Java server side technology, i.e. Java Servlets and JSP.

Applications also send and receive data to and from the technology adapters using MDML and HTTP. To minimise the effort required to generate MDML from within application code the framework includes a Java based MDML API. However, as MDML is an open specification, it is also possible develop MDML compliant applications using in-house code or development environments other than Java.

Adaption

Each technology adapter in the adaption layer converts between a specific device technology and the micro device protocol. A technology adaptor has the following core functions:

- Provision of a receiver and transmitter giving access to the underlying network transport mechanism for the device
- Translation of incoming messages into MDML requests that are sent to the access broker via HTTP 'post' messages
- Translation of MDML responses from the access broker into the format used by the underlying network transport mechanism
- Management of user and device context – this enables users to enter the minimum number of keystrokes required to access the services they require.

The structure of a technology adapter is shown in Fig. 3. While, currently, the capabilities of the micro devices are comparatively restrictive, as new technologies become available, the component-based architecture of the framework will allow new technology adaptors to be included.

Access

The access broker:

- Brokers user access to content and applications for 'pull' type operations
- Brokers application access to user's micro devices for 'push' type operations

This allows users, service providers, content providers and application developers to specify which services are available to which users and on which devices. Both pre-paid and post-paid billing for content and services is also managed by the access broker.

Fig. 4 shows how a request from a technology adaptor is dealt with by the access broker. On receiving a request, the access broker:

- Identifies the associated application
- Checks user subscription to the application
- Performs any pre-access checks and debits the user's account if necessary
- Forwards the request to the application, performing any re-routing and URL translation required

On receiving a response, the access broker:

- Parses the message
- Performs any billing or charging on the panel
- Generates advertising content based on the application information and the panel contents
- Determines where individual components of the message should be sent depending on the available device adaptors
- Assembles and sends the individual message components to the appropriate adapters

Applications

Applications are defined as a set of MDML pages with associated logic, image and sound files. An application is defined in an application configuration file that specifies the following:

- The application name
- The style-sheets and device configurations associated with the application
- The locations of the application's associated files
- Its billing capabilities
- Images and icons used to identify the application to the user (e.g. on a personalisation web page or in a PDA folder)

An application's files reside in a directory hierarchy that can be packaged up and deployed in a new location without any porting constraints.

Fig. 5 shows the many ways that applications and content can be developed using the Framework. An application consists of both MDML content and logic. The logic resides in a series of Java Beans that are invoked automatically when a given action is triggered by the user. A standard mapping mechanism is used to ensure that all actions either link to a new panel or are handled by a Java Bean. This allows an application developer to simulate dynamic behaviour using either static panels, JSP or both and then transparently migrate to a more interactive solution without any re-deployment problems.

Micro Device Mark-up Language

MDML provides a simple abstract mechanism for defining mobile data to be presented to a user over a variety of devices. It is a 'content only' format – formatting and styling have been excluded deliberately in order to provide a definition that is

both device and protocol independent. MDML allows a content editor (or a dynamic content generator) to:

- Define content for micro devices in a hierarchical fashion
- Associate actions and links with content
- Define advertising related instructions for content
- Specify billing mechanism for content

Since MDML is an XML-compliant data format, it is possible to translate XML data feeds into MDML using standard style-sheet technology. It is also possible to translate MDML into HTML and WML using style-sheets.

Content Definition

MDML allows editors to create simple text-based content. Editors can also associate a set of images and sound files with the content. The root element of all MDML content is a panel. A panel can be regarded as an abstraction of the screen on the mobile device. A panel contains one of the following elements:

- **Text** – Character data arranged in paragraphs
- **Menu** – This defines a menu of items from which the user can select
- **List** – This defines an ordered list of items which may or may not be selectable
- **Form** – This is a panel through which the user can enter information
- **Table** – A tabular set of information

A panel element can be associated with sound and image elements. Sound elements can be defined as either short duration sound snippets that can be transferred to the micro device over standard communication links or sound streams that must be accessed by a different mechanism such as a streamed voice connection or an mp3

device. Image elements are defined in an image set. An image set is a set of image files containing different versions of the same image. These versions may be in different encoding formats (e.g. GIF or WBMP) and of different sizes. They each have a set of attributes that define:

- The encoding format
- The image size
- The allowable transformations for this version of the image

This provides a mechanism that allows different images to be sent to different devices, thus enabling the content editor to control the appearance of the image precisely.

Linking and Actions

Elements can also have actions and links as attributes. An action attribute defines an action that can be performed on the element (e.g. selection). A link attribute defines another MDML panel to which control is transferred automatically when it is triggered. A panel can also define action elements. Action elements are triggered by the user in some device-specific manner. As with an action attribute, an action element may have an associated link that is accessed automatically when the action is trigged.

Advertising Support

A panel can have associated with it a set of elements defining the advertising context of the panel. These define whether advertising is supported for the panel and either define a link from which advertising content can be extracted (e.g. a sponsor's message) or else define categories which can be used to determine advertising for the panel.

Content Billing

MDML defines elements that determine how billing is performed on the panel. Editors can define the type of billing for a panel and can define values for the billing itself. Billing elements can contain either a numeric value or a category. A numeric value corresponds directly to the amount that should be debited from the users account. A category (e.g. PREMIUM or SPORT) determines which user profile the user must belong to in order to access the content.

It will be appreciated that the applications framework 1 supports device-to-server interaction in a versatile and stable manner. It allows a high degree of user interactivity, and also excellent personalisation because of storage of user contexts. The framework also allows use of a wide range of access devices. Another important advantage is that adaptation is based on the actual features of the access device technology so that there is effectively "seamless" access to framework content from different devices. Regarding the content made available through the framework, this includes interactive applications and dynamic content using JSP and MicroWeb APIs. There is excellent content brokering and attribute-based routing as well as enhanced advertising and billing capabilities.

The invention is not limited to the embodiments described but may be varied in construction and detail.

Claims

1. An applications framework comprising:

an adapter comprising means for interfacing with a user mobile device;

an access broker comprising means for routing requests and responses according to device technology and content attributes; and

an application platform comprising means for allowing access to content and application servers via the access broker.

2. An applications framework as claimed in claim 1, wherein the adapter comprises means for managing the context for a device making a request, and for converting an input into a Web request using input data, device context, and application context.

3. An applications framework as claimed in claim 2, wherein the adapter comprises means for adding user and location context to requests.

4. An applications framework as claimed in claim 2 or 3, wherein the adapter comprises means for translating responses into device-specific format using response date, device context, and application context.

5. An applications framework as claimed in any of claims 2 to 4, wherein the adapter comprises means for updating and storing context between device interactions.

6. An applications framework as claimed in any preceding claim, wherein the framework comprises a plurality of adapters, each associated with a type of mobile device.

7. An applications framework as claimed in any preceding claim, wherein the access broker comprises means for controlling access to Web applications according to user subscription, in which responses are split up and routed according to adapter capabilities, content attributes, and user-specified rules.

8. An applications framework as claimed in any preceding claim, wherein the access broker comprises means for managing a register of adapter capabilities and of currently accessible adapters for each user.

9. An applications framework as claimed in any preceding claim, wherein the access broker comprises means for translating service values placed by applications, and for routing the translated data to external systems.

10. An applications framework as claimed in any preceding claim, wherein the application platform comprises means for an API to allow alternative interfaces for interactive applications.

11. An applications framework as claimed in any preceding claim, wherein the adapter, the access broker, and the application platform comprises means for communicating with each other using an XML-compliant markup language.

12. An applications framework as claimed in claim 11, wherein content is defined in elements, in which a root element is an abstraction of a mobile device screen.

13. An applications framework as claimed in claim 12, wherein sound streams and images are defined as elements.

14. A computer program product comprising software code for completing an applications framework as claimed in any preceding claim when executing on a digital computer.

ABSTRACT

"An applications framework for network communication"

A framework (1) has adapters (3) which interface with mobile devices and manages context for them. An access broker (2) controls access to framework-based applications and other functions such as billing. An applications platform (4) provides request/response mechanisms for access to applications on external servers.

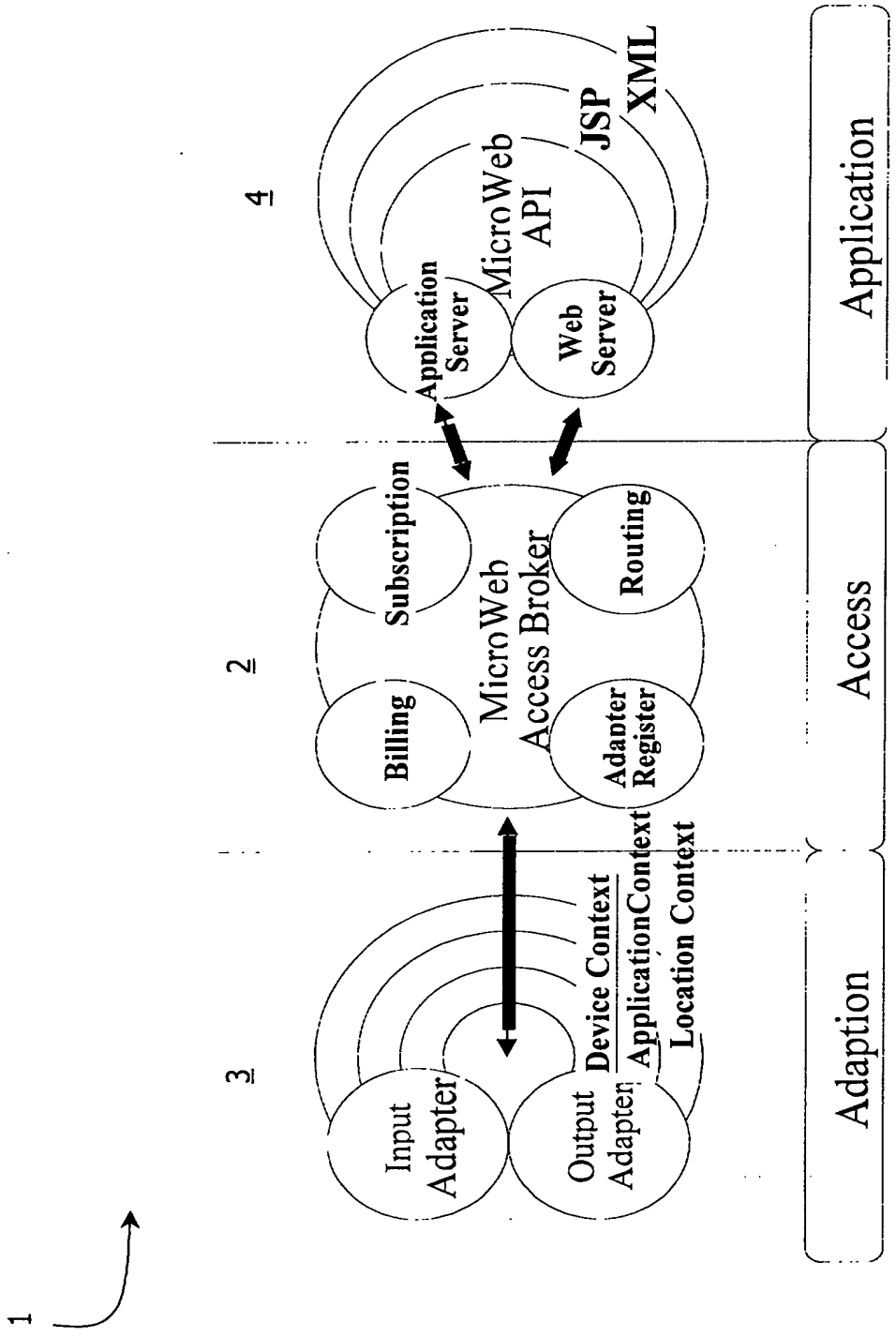
1
3
2
4
Input Adapter
Output Adapter
Device Context
ApplicationContext
Location Context
Billing
Subscription
MicroWeb Access Broker
Adanter Register
Routing
Application Server
Web Server
MicroWeb API
JSP
XML
Adaption
Access
Application

Fig. 1

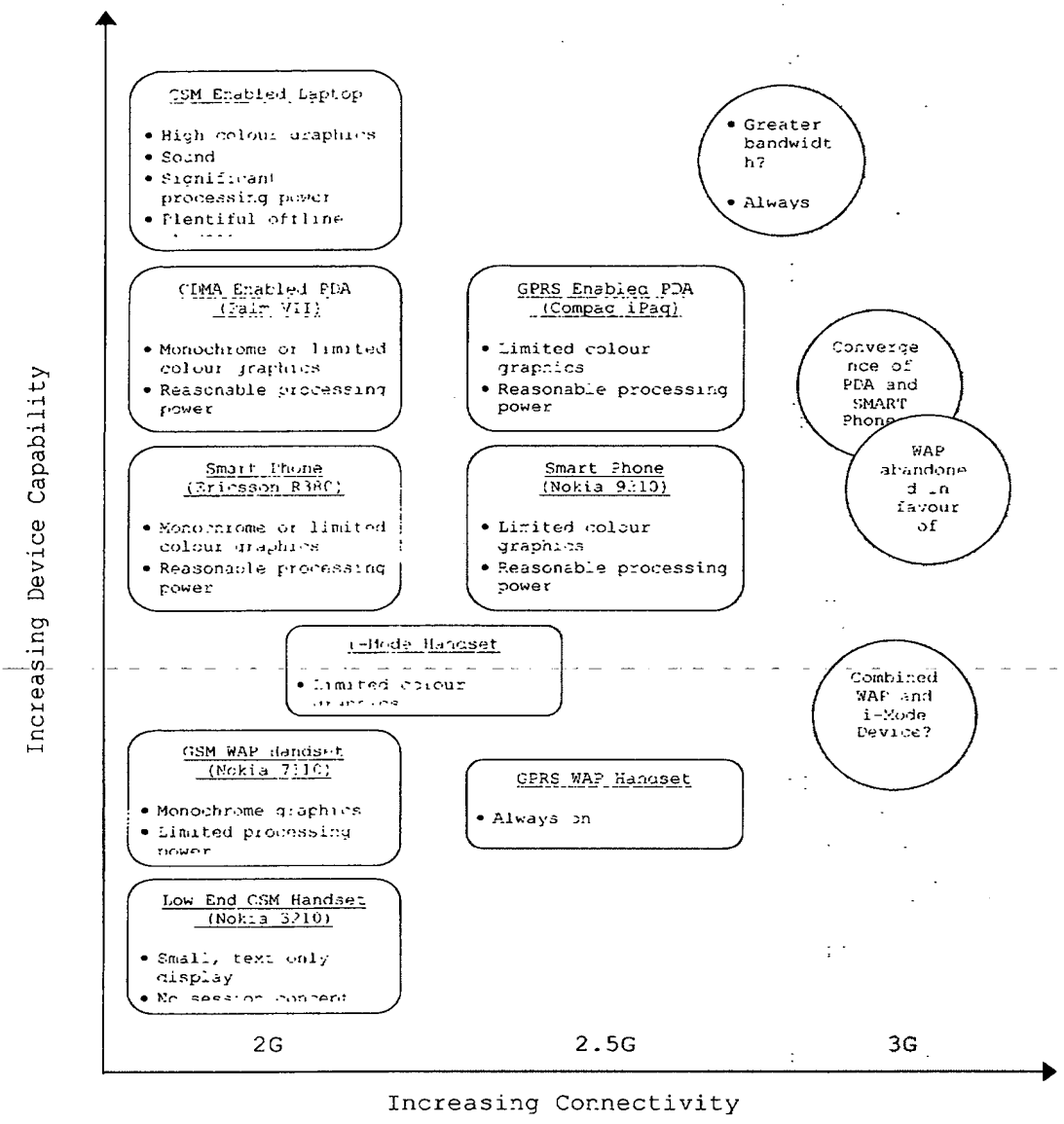
GSM Enabled Laptop
• High colour graphics
• Sound
• Significant processing power
• Plentiful offline
• Greater bandwidth?
• Always
CDMA Enabled PDA (Palm VII)
• Monochrome or limited colour graphics
• Reasonable processing power
GPRS Enabled PDA (Compaq iPaq)
• Limited colour graphics
• Reasonable processing power
Convergence of PDA and SMART Phone
WAP abandoned in favour of
Smart Phone (Ericsson R380)
• Monochrome or limited colour graphics
• Reasonable processing power
Smart Phone (Nokia 9210)
• Limited colour graphics
• Reasonable processing power
i-Mode Handset
• Limited colour
Combined WAP and i-Mode Device?
GSM WAP Handset (Nokia 7110)
• Monochrome graphics
• Limited processing power
GPRS WAP Handset
• Always on
Low End GSM Handset (Nokia 3210)
• Small, text only display
2G
2.5G
3G
Increasing Device Capability
Increasing Connectivity

Fig. 2

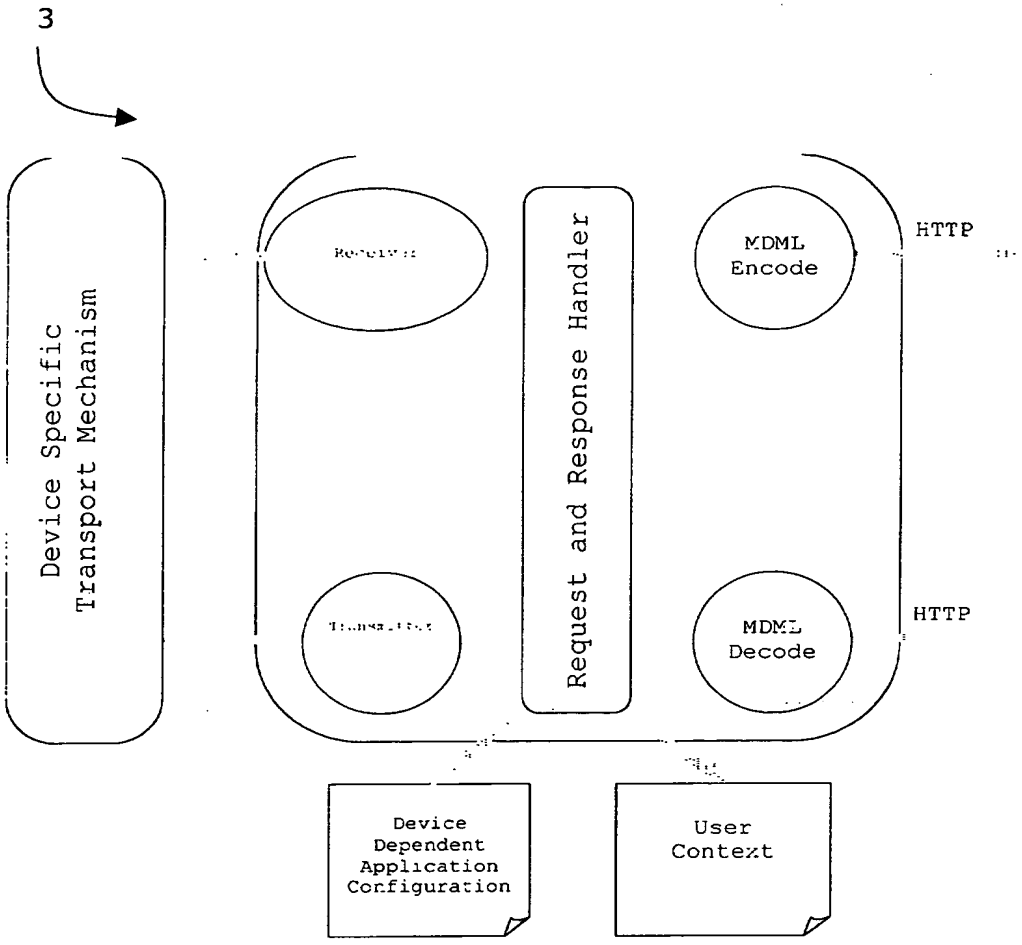
3
Device Specific Transport Mechanism
Request and Response Handler
MDML Encode
MDML Decode
HTTP
HTTP
Device Dependent Application Configuration
User Context

Fig. 3

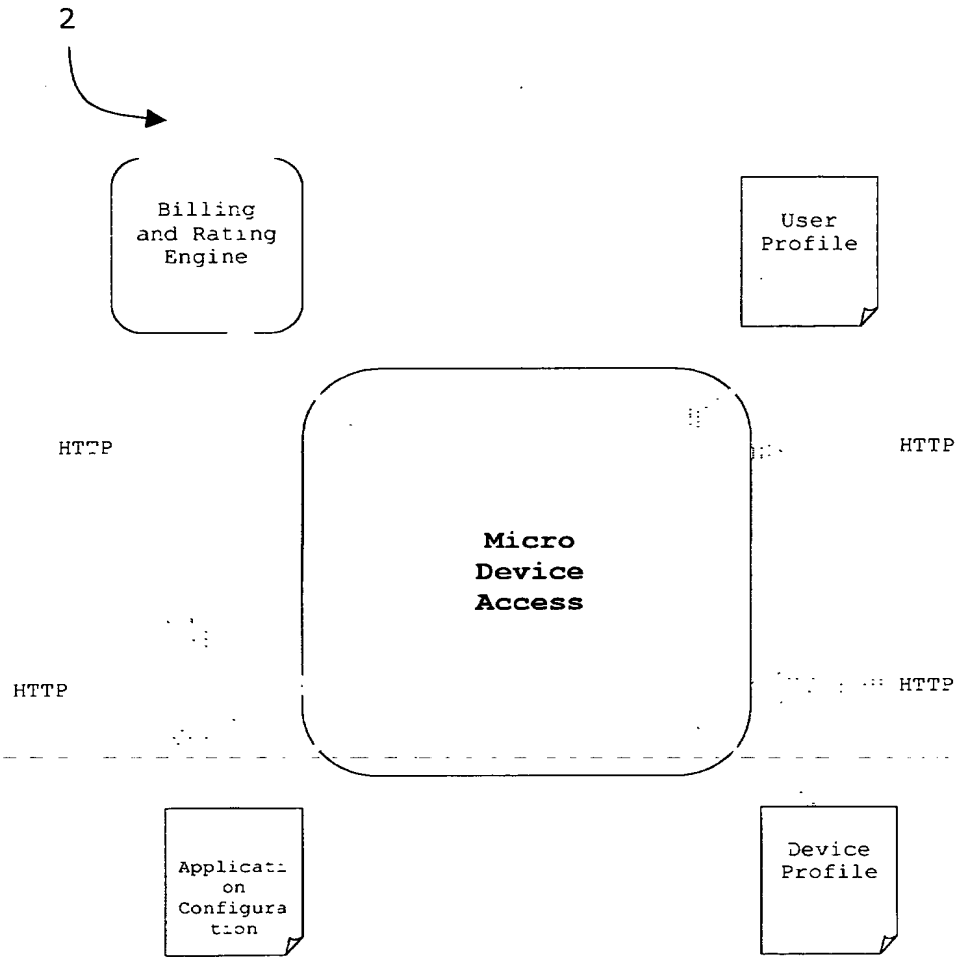
2
Billing and Rating Engine
User Profile
HTTP
HTTP
Micro Device Access
HTTP
HTTP
Application Configuration
Device Profile

Fig. 4

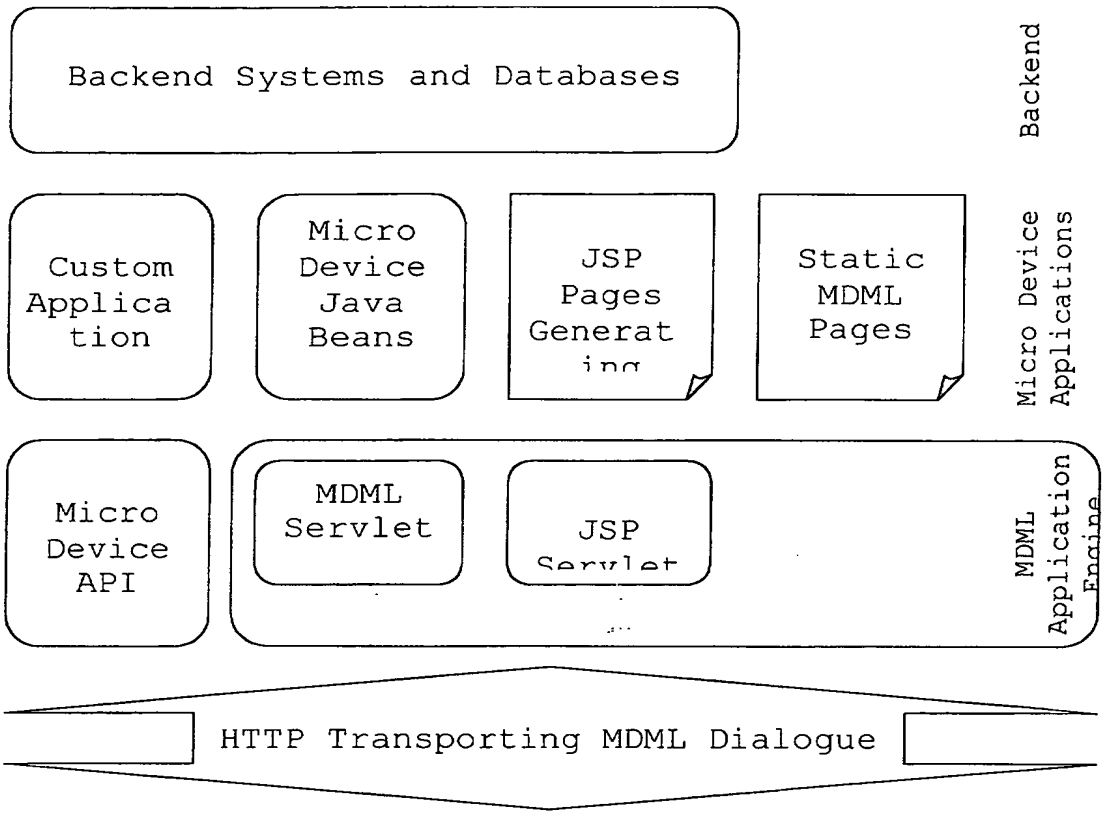
Backend Systems and Databases
Backend
Custom Application
Micro Device Java Beans
JSP Pages Generating
Static MDML Pages
Micro Device Applications
Micro Device API
MDML Servlet
JSP Servlet
MDML Application Engine
HTTP Transporting MDML Dialogue

Fig. 5